www.ijecs.in

International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 6 June, 2014 Page No. 6688-6691

Performance Analysis of RSA Algorithm Using Elliptic Curve Cryptography, Diffie Hellman And OAEP

Neha Garg, Partibha Yadav
Student, CSE Department
PDMCEW MDU Rohtak, India
Assistant Professor, CSE Department PDMCEW
Bahadurgarh, India
nehagarg9115@gmail.com
pratibha1007@gmail.com

Abstract:- Communication is the very important part of any type of network for making it possible to transfer data from one network to another. Communication needs quality and security for better performance which provides confidentiality to users. Cryptographic technique is one of the principal means to protect information security. It is to ensure the information confidential and also provides digital signature, authentication, secret sub-storage, system security and other functions. Therefore, the encryption and decryption solution can ensure the confidentiality of the information, as well as the integrity of information to prevent information from unauthorized access and counterfeiting. Encryption and decryption algorithm's security is depends on the algorithm and also depends on the key confidentiality. In this paper we compare asymmetric key cryptography algorithm and also implement these algorithm. we compare these algorithm on the basis on time.

Keywords:- Cryptography, RSA algorithm, Encryption, Decryption, Elliptic curve cryptography.

I. INTRODUCTION

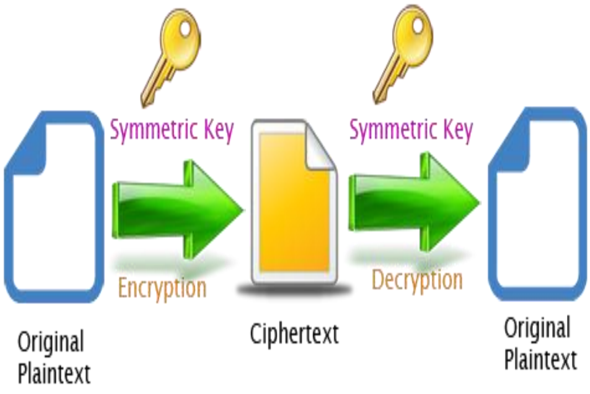
Cryptography as the discipline that studies the mathematical techniques related to Information security such as providing the security services of confidentiality, data integrity, authentication and non repudiation. In cryptographic terminology, the message is called plaintext. Encoding the contents of the message in such a way that its contents cannot be unveiled by outsiders is called encryption. The encrypted message is called the cipher text. The process of retrieving the plaintext from the cipher text is called decryption. Cryptography falls into two important categories: 1. Symmetric-key cryptography or secret key refers to encryption methods in which both the sender and receiver share the same key. Symmetric key ciphers are implemented as either block ciphers or stream ciphers. A block cipher enciphers input in blocks of plaintext as opposed to individual characters, the input form used by a stream cipher. 2. Asymmetric key Cryptography or public key refers to a cryptographic system requiring two separate keys, one of which is secret and one of which is public. It offers so many advantages:

1. **Confidentiality:** It guarantees that the sensitive information can only be accessed
by those users/entities authorized to unveil it
2. **Data integrity:** It is a service which addresses the unauthorized alteration of data. This property refers to data that has not been changed, destroyed, or lost in a malicious or accidental manner.
3. **Authentication:** It is a service related to identification. This function applies to both entities and information itself. Two parties entering into a communication should identify each other.
4. **Non-repudiation:** It is a service which prevents an entity from denying previous commitments or actions.

The risk of intrusion and eavesdropping goes up as electronic communication equipment becomes increasingly wireless and ubiquitous. With the specter of hackers/crackers looming, security is becoming a major consideration in a growing number of embedded systems. In this paper we focus asymmetric algorithm like (RSA, Elliptic curve, Diffie Hellman and OAEP).

II. SYMTERIC KEY CRYPTOGRAPHY

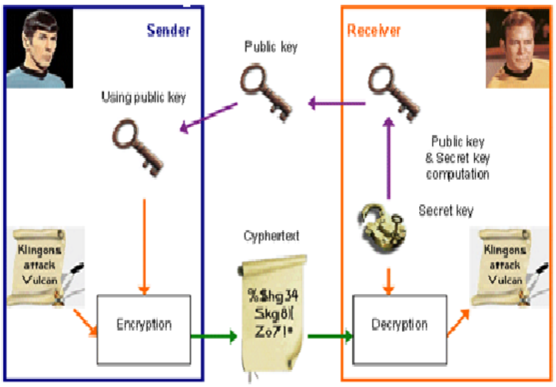
Symmetric encryption is the oldest and best-known technique. A secret key, which can be a number, a word, or just a string of random letters, is applied to the text of a message to change the content in a particular way. This might be as simple as shifting each letter by a number of places in the alphabet. As long as both sender and recipient know the secret key, they can encrypt and decrypt all messages that use this key.



III. ASYMMETRIC KEY CRYPTOGRAPHY

In Asymmetric cryptography a pair of keys is used to encrypt and decrypt a message so that it is transmitted securely. Initially, a network user receives a public and private key pair from a Certificate Authority. The process of encryption using asymmetric cryptography can be explained by following steps -

- Use a key (public key) to encrypt a message.
- Another (private key) to decrypt a message.
- Private Key known to owner and used only by owner.



IV. RSA ALGORITHM

RSA is a cryptosystem, which is known as one of the first practicable public-key cryptosystems and is widely used for secure data transmission. In RSA, this asymmetry is based on the practical difficulty of factoring the product of two large prime numbers, the factoring problem. RSA stands for Ronald Rivest, Adi Shamir and Leonard Adleman, who first publicly described the algorithm in 1977. It involves three steps: key generation, encryption and decryption It is still widely used in electronic commerce protocols.

RSA algorithm for generating key:

1. Choose two distinct prime numbers *p* and *q*.(integers *p* and *q* should be chosen at random, and should be of similar bit-length eg.1024 bits)
2. Compute *n* = *p***q*.(*n* is used as the modulus for both the public and private keys)
3. Select the public key (i.e. encryption key) E such that it is not a factor of (P – 1) and (Q -1).
4. Select the private key (i.e. the decryption key) D such that the following equation is true (D x E) mod (P – 1) x (Q – 1) = 1.
5. For encryption, calculate the cipher text CT from the plain text PT as follows: CT = PTE mod N.
6. Then send CT as the cipher text to the receiver.
7. For decryption, calculate the plain text PT from the cipher text CT as follows: PT = CTD mod N.

RSA involves a public key and a private key. The public key can be known by everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted in a reasonable amount of time using the private key. The main feature of RSA algorithm is the selection of large prime number (p, q) because it is logical that fraction of large number is always typical and any users or force attackers could not be able to find the capable numbers, timely to force attack is shortly non-feasible.

V. ELLIPTIC CURVE CRYPTOGRAPHY

An elliptic curve is given by an equation in the form of $y2 = x3 + ax + b$ where $4a3 + 27b2 \neq 0$.
Many interesting problems arise from the set of points on elliptic curves over a finite field under group operations. The finite fields that are commonly used are those over primes (F*p*) and binary fields ($_{F2}{}^{n}$). The security of ECC is based on the elliptic curve discrete logarithm problem (ECDLP). This problem is defined as:
Given points X, Y on the elliptic curve, find *z* such that:
X = *z*Y
The discrete logarithm problem over this group in a finite field is a good one way function because there are currently no known polynomial time attacks for solving the problem.
ECC was developed independently by Neal Koblitz and Victor Miller in 1985.

Key Generation:-

To generate a public and private key pair for use in ECC communications, an entity would perform the following steps:
1. Find an elliptic curve *E(K)*, where *K* is a finite field such as F*p* or $_{F2}{}^{n}$, and a find point *Q* on *E(K)*. *n* is the order of *Q*. Recommended domain parameters for *E*(*K*) are suggested in.
2. Select a pseudo random number *x* such that $1 \leq x \leq (n - 1)$.
3. Compute point *P* = *xQ*.
4. Your ECC key pair is (*P*, *x*), where *P* is your public key, and *x* is your private key.

VI. DIFFIE HELLMAN

Diffie-Hellman key exchange, also called exponential key exchange, is a method of digital encryption that uses numbers raised to specific powers to produce decryption keys on the basis of components that are never directly transmitted, making the task of a would-be code breaker mathematically overwhelming. To implement Diffie-Hellman, the two end users Alice and Bob, while communicating over a channel they know to be private, mutually agree on positive whole numbers *p* and *q*, such that *p* is a prime number and *q* is a generator of *p*. The generator *q* is a number that, when raised to positive whole-number powers less than *p*, never produces the same result for any two such whole numbers. The value of *p* may be large but the value of *q* is usually small.

1. Taking two numbers “P” and “G” “P” is a large prime number “G” is called the base.
2. Picks a secret number “A” as first secret number = A, then picks another secret number “B” as second secret number = B.
3. Computes first public number X = GA mod P, and public number = X. Then computes second public number Y = GB mod P, and public number = Y.
4. Exchange their public numbers.
5. First knows P, G, A, X, Y, Second knows P, G, B, X, Y.
6. Computes First session key as KA = YA mod P OR KA = (GB mod P)A mod P OR KA = (GB) A mod P OR KA = GBA mod P.
7. Computes second session key as KB = XB mod P OR KB = (GA mod P)B mod P OR KB = (GA) B mod P OR KB = GAB mod P.
8. Fortunately for Both by the laws of algebra, First session key “KA” is the same as Second session key “KB”, or KA = KB = K.
9. Know we have both the secret value as “K”.

VII. OAEP(OPTIMAL ASYMMETRIC ENCRYPTION PADDING)

Optimal Asymmetric Encryption Padding (OAEP) is a padding scheme often used together with RSA encryption. OAEP was introduced by Bellare and Rogaway, and subsequently standardized in PKCS #1v2 and RFC 2437.The OAEP algorithm is a form of Feistel network which uses a pair of random oracles G and H to process the plaintext prior to asymmetric encryption. When combined with any secure trapdoor one-way permutation f, this processing is proved in the random oracle model to result in a combined scheme which is semantically secure under chosen plaintext attack (IND-CPA). When implemented with certain trapdoor permutations (e.g., RSA), OAEP is also proved secure against chosen cipher text attack. OAEP can be used to build and all-or-nothing transform.

OAEP satisfies the following two goals:

1. Add an element of randomness which can be used to convert a deterministic encryption scheme (e.g., traditional RSA) into a probabilistic scheme.
2. Prevent partial decryption of cipher texts (or other information leakage) by ensuring that an adversary cannot recover any portion of the plaintext without being able to invert the trapdoor one-way permutation f.

The original version of OAEP (Bellare/Rogaway, 1994) showed a form of "plaintext awareness" (which they claimed implies security against chosen cipher text attack) in the random oracle model when OAEP is used with any trapdoor permutation. Subsequent results contradicted this claim, showing that OAEP was only IND-CCA1 secure. However, the original scheme was proved in the random oracle model to be IND-CCA2 secure when OAEP is used with the RSA permutation using standard encryption exponents, as in the case of RSA-OAEP. An improved scheme (called OAEP+) that works with any trapdoor one-way permutation was offered by Victor Shoup to solve this problem. More recent work has shown that in the standard model (that is, when hash functions are not modelled as random oracles), that it is impossible to prove the IND-CCA2 security of RSA-OAEP under the assumed hardness of the RSA problem

VIII. COMPARISION OF THESE ALGORITHM

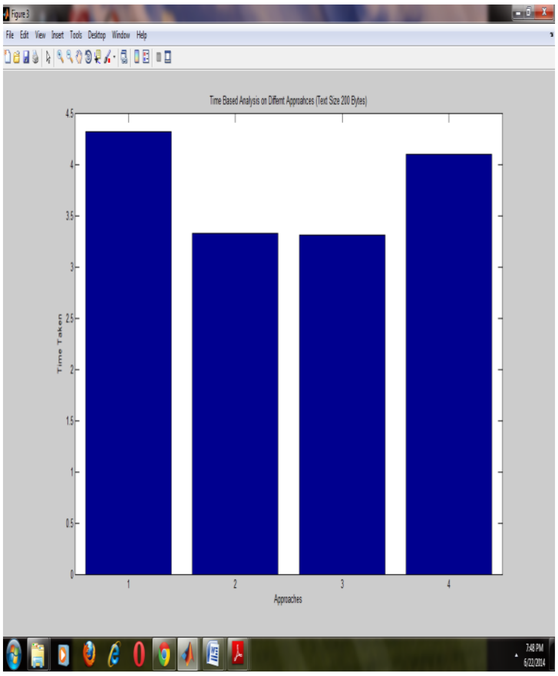


Figure 3
File Edit View Insert Tools Desktop Window Help
Time Based Analysis on Diffent Approahces (Text Size 200 Bytes)
Time Taken
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
1
2
3
4
Approaches
7:48 PM
6/22/2014

In this we compare four asymmetric algorithms on the basis of time. In this we implement all these algorithm and find out one which takes same data for security purpose. In this we find optimal asymmetric algorithm is best approach for confidentiality purpose because it takes less time to execute and also provide best security. It is enhanced of RSA algorithm but RSA takes more time in execution and not much secure. Elliptic curve cryptography and RSA both provide same security. it becomes more difficult to decrypt the private key generated by OAEP. Using diffie hellman key exchange algorithm decryption is more difficult. The RSA provides highest security to the business application. When comparing with RSA, RSA – OAEP algorithm requires more time for encryption decryption. Whereas RSA-OAEP is more secured cryptography algorithm than RSA, because RSA –OEAP includes OAEP concept, which is more difficulty for the intruder to find the plain text from the encrypted message.

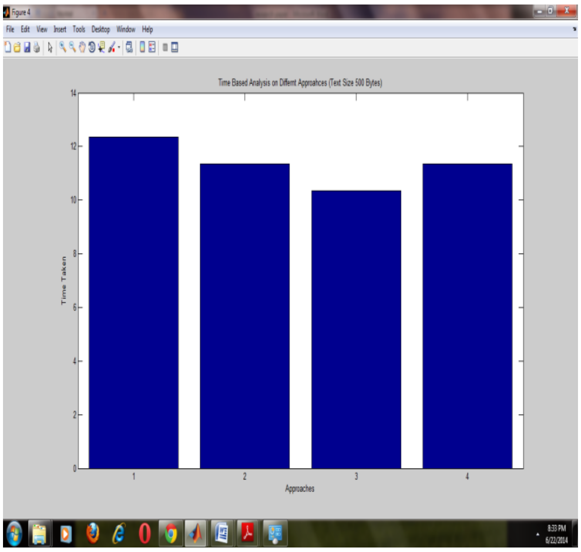


Figure 4
File Edit View Insert Tools Desktop Window Help
Time Based Analysis on Differnt Approahces (Text Size 500 Bytes)
Time Taken
14
12
10
8
6
4
2
0
1
2
3
4
Approaches
8:33 PM
6/22/2014

When we take different size of data then performance is different which shows in figures. But all time our best approach is OAEP and second is Elliptic curve cryptography. Both approaches are good for securing of data.

IX. CONCLUSION AND FUTURE WORK

Asymmetric algorithm can be used to eliminate the problem of user, when a users transmit the data over the network there is no guaranteed that data is original data or not. It means any unauthorized person can easily access that data and also they can alter that data. Asymmetric key is used for providing security to the users when they transmit data over the network. Public key cryptography uses two keys one for encryption and other for decryption so its provide better security for users. RSA algorithm is providing much overhead in encrypting the text. When we compare the elliptic curve cryptography with RSA then we identify that ECC provides less overhead compare to the RSA. In case of encrypting the text ECC is better than RSA. OAEP is a padding scheme which is used by RSA algorithm. OAEP provides the better security compared to RSA. In RSA algorithm, message length should be less than the bit length so unauthorized person can easily crack it but in case of OAEP there is no problem with the message length because it takes large bit length. In OAEP speed of encryption process is better than RSA. RSA provides highest security to the business application so this scheme can be used for encryption of long messages without employing the hybrid and symmetric encryption. RSA key generation is significantly slower than ECC key generation for RSA key of sizes 1024 bits and greater. Purpose of this paper is to find out the best algorithm which provides the security to users.

X. REFERENCE

[1]. Jiezhao Peng, Qi Wu, Research and Implementation of RSA Algorithm in Java, 978-0-7695-3366-7/08 $25.00 © 2008 IEEE

[2]. Xin Zhou, Xiaofei Tang, Research and Implementation of RSA Algorithm for Encryption and Decryption, 978-1-4577-0399-7/111$26.00 ©20111EEE

[3]. Li Dongjiang, Wang Yandan and Chen Hong, The research on key generation in RSA public- key cryptosystem, 2012 Fourth International Conference on Computational and Information Sciences, 978-0-7695-4789-3/12 $26.00 © 2012 IEEE

[4]. Shilpi Gupta, Jaya Sharma, A Hybrid Encryption Algorithm based on RSA and Diffie-Hellman, 978-1-4673-1344-5/12/$31.00 ©2012 IEEE

[5]. Alese, B. K., Philemon E. D., Falaki and S. O., Comparative Analysis of Public-Key Encryption Schemes, International Journal of Engineering and Technology Volume 2 No. 9, September, 2012

[6]. Aqeel Khalique, Kuldip Singh and Sandeep Sood, Implementation of Elliptic Curve Digital Signature Algorithm, *International Journal of Computer Applications (0975 – 8887) Volume 2 – No.2, May 2010*

[7]. Ram Ratan Ahirwal and Manoj Ahke, Elliptic Curve Diffie-Hellman Key Exchange Algorithm for Securing Hypertext Information on Wide Area Network, International Journal of Computer Science and Information Technologies, Vol. 4 (2) , 2013, 363 – 368

[8]. Vishal Garg and Rishu, Improved Diffie-Hellman Algorithm for Network Security Enhancement, Int.J.Computer Technology & Applications,Vol 3 (4), 1327-1331, July-August 2012

[9]. Rounak Sinha, Hemant Kumar Srivastava and Sumita Gupta, Performance Based Comparison Study of RSA and Elliptic Curve Cryptography, International Journal of Scientific & Engineering Research, Volume 4, Issue 5, May-2013 ,ISSN 2229-5518

[10]. M. Preetha, M. Nithya, A STUDY AND PERFORMANCE
ANALYSIS OF RSA ALGORITHM, *IJCSMC, Vol. 2, Issue. 6, June 2013, pg.126 – 139*

[11]. Ashish Vijay,Priyanka Trikhaand Kapil Madhur, A New Variant of RSA Digital Signature, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 10, October 2012

[12]. Botes, J.J., Penzhorn, W.T., 1994. An implementation of an elliptic curve cryptosystem.